Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently amended) A method, in a data processing system, for processing instructions of a computer program, comprising:

associating a performance indicator with at least one instruction of a portion of code of the computer program;

enabling counting, by a processor, of a number of times instructions of the portion of code of the computer program, having an associated performance indicator, are executed to generate a first count;

enabling counting, by a processor, of a number of times there is a cache miss when executing instructions of the portion of code of the computer program having associated performance indicators to generate a second count; [[and]]

determining if a problem condition is present in a cache based on the first count and the second count;

wherein if the problem condition is determined to be present, setting a control bit in the processor indicating that a chase tail operation is to be performed with reload operations of a cache; and

wherein if the problem condition is determined to be absent, terminating the processing of the at least one instruction.

2. (Original) The method of claim 1, wherein the first count and the second count are maintained in a hardware counter associated with the processor.

3. (Original) The method of claim 1, wherein determining if a problem condition is present in a cache based on the first count and the second count includes:

generating a cache hit-miss ratio based on the first count and the second count;

comparing the cache hit-miss ratio to a predetermined threshold value; and

determining that a problem condition exits if a predetermined relationship between the cache hit-miss ratio and the predetermined threshold value is present.

4. (Original) The method of claim 1, further comprising:

sending an interrupt to an interrupt handler of a performance monitoring application if a problem condition is determined to be present.

5. (Original) The method of claim 3, wherein the predetermined relationship is that the cache hit-miss ration meets or falls below the predetermined threshold value.

6. (Original) The method of claim 1, wherein if a problem condition is determined to be present, call flow support is initiated comprising:

determining if there is sufficient capacity in the cache to load the portion of code without overwriting existing entries in the cache; and

loading the portion of code to a dedicated storage area if there is not sufficient capacity in the cache.

7. (Canceled)

8. (Currently amended) The method of claim [[7]] 1, wherein the chase tail operation includes:

determining if there is sufficient capacity in the cache to load the portion of code without overwriting existing entries in the cache; and

loading the portion of code to a dedicated storage area if there is not sufficient capacity in the cache.

9. (Currently amended) The method of claim [[7]] 1, wherein the chase tail operation includes:

determining if there is sufficient capacity in the cache to load the portion of code without overwriting existing entries in the cache;

identifying an overflow portion of the portion of code that cannot be loaded into the cache without overwriting existing entries in the cache;

identifying a non-overflow portion of the portion of code that can be loaded into the cache without overwriting existing entries in the cache;

loading the non-overflow portion of the portion of code into the cache; and

loading the overflow portion of the portion of code into a dedicated storage area.

10. (Original) The method of claim 6, wherein the dedicated storage area is a reserved portion of the cache.

11. (Original) The method of claim 8, wherein the dedicated storage area is a reserved portion of the cache.

12. (Original) The method of claim 6, wherein the call flow support further includes:

associating a performance indicator with the portion of code indicating that, upon execution of instructions in the portion of code, the instructions are to be retrieved from the dedicated storage area.

13. (Original) The method of claim 8, wherein the chase tail operation further includes:

associating a performance indicator with the portion of code indicating that, upon execution of instructions in the portion of code, the instructions are to be retrieved from the dedicated storage area.

14. (Original) The method of claim 9, wherein the chase tail operation further includes:

associating a performance indicator with the overflow portion of the portion of code indicating that, upon execution of instructions in the overflow portion of the portion of code, the instructions are to be retrieved from the dedicated storage area.

15. (Currently amended) A computer program product in a recordable-type computer readable medium for processing instructions of a computer program, comprising:

first instructions for associating a performance indicator with at least one instruction of a portion of code of the computer program;

second instructions for enabling counting, by a processor, of a number of times instructions of the portion of code of the computer program, having an associated performance indicator, are executed to generate a first count;

third instructions for enabling counting, by a processor, of a number of times there is a cache miss when executing instructions of the portion of code of the computer program having associated performance indicators to generate a second count; [[and]]

fourth instructions for determining if a problem condition is present in a cache based on the first count and the second count;

fifth instructions for setting a control bit in the processor indicating that a chase tail operation is to be performed with reload operations of a cache, if a problem condition is determined to be present; and

sixth instructions for terminating the processing of the at least one instruction, if the problem condition is determined to be absent.

16. (Original) The computer program product of claim 15, wherein the first count and the second count are maintained in a hardware counter associated with the processor.

17. (Original) The computer program product of claim 15, wherein the fourth instructions for determining if a problem condition is present in a cache based on the first count and the second count include:

instructions for generating a cache hit-miss ratio based on the first count and the second count;

instructions for comparing the cache hit-miss ratio to a predetermined threshold value; and

instructions for determining that a problem condition exits if a predetermined relationship between the cache hit-miss ratio and the predetermined threshold value is present.

18. (Currently amended) The computer program product of claim 15, further comprising:

[[fifth]] seventh instructions for sending an interrupt to an interrupt handler of a performance monitoring application if a problem condition is determined to be present.

19. (Original) The computer program product of claim 17, wherein the predetermined relationship is that the cache hit-miss ration meets or falls below the predetermined threshold value.

20. (Original) The computer program product of claim 15, wherein if a problem condition is determined to be present, call flow support is initiated comprising:

instructions for determining if there is sufficient capacity in the cache to load the portion of code without overwriting existing entries in the cache; and

instructions for loading the portion of code to a dedicated storage area if there is not sufficient capacity in the cache.

21. (Canceled)

22. (Currently amended) The computer program product of claim [[21]] 15, wherein the chase tail operation includes:

instructions for determining if there is sufficient capacity in the cache to load the portion of code without overwriting existing entries in the cache; and

instructions for loading the portion of code to a dedicated storage area if there is not sufficient capacity in the cache.

23. (Currently amended) The computer program product of claim [[21]] 15, wherein the chase tail operation includes:

instructions for determining if there is sufficient capacity in the cache to load the portion of code without overwriting existing entries in the cache;

instructions for identifying an overflow portion of the portion of code that cannot be loaded into the cache without overwriting existing entries in the cache;

instructions for identifying a non-overflow portion of the portion of code that can be loaded into the cache without overwriting existing entries in the cache;

instructions for loading the non-overflow portion of the portion of code into the cache; and

instructions for loading the overflow portion of the portion of code into a dedicated storage area.

24. (Original) The computer program product of claim 20, wherein the dedicated storage area is a reserved portion of the cache.

25. (Original) The computer program product of claim 22, wherein the dedicated storage area is a reserved portion of the cache.

26. (Original) The computer program product of claim 20, wherein the call flow support further includes:

instructions for associating a performance indicator with the portion of code indicating that, upon execution of instructions in the portion of code, the instructions are to be retrieved from the dedicated storage area.

27. (Original) The computer program product of claim 22, wherein the chase tail operation further includes:

instructions for associating a performance indicator with the portion of code indicating that, upon execution of instructions in the portion of code, the instructions are to be retrieved from the dedicated storage area.

28. (Original) The computer program product of claim 23, wherein the chase tail operation further includes:

instructions for associating a performance indicator with the overflow portion of the portion of code indicating that, upon execution of instructions in the overflow portion of the portion of code, the instructions are to be retrieved from the dedicated storage area.

29. (Currently amended) An apparatus for processing instructions of a computer program, comprising:

means for associating a performance indicator with at least one instruction of a portion of code of the computer program;

means for enabling counting, by a processor, of a number of times instructions of the portion of code of the computer program, having an associated performance indicator, are executed to generate a first count;

means for enabling counting, by a processor, of a number of times there is a cache miss when executing instructions of the portion of code of the computer program having associated performance indicators to generate a second count; [[and]]

means for determining if a problem condition is present in a cache based on the first count and the second count;

wherein if a problem condition is determined to be present, means for setting a control bit in the processor indicating that a chase tail operation is to be performed with reload operations of a cache; and

wherein if the problem condition is determined to be absent, means for terminating the processing of the at least one instruction.